THE DELTA PI EPSILON JOURNAL

DEVOTED TO
PROFESSIONAL
BUSINESS
EDUCATION

ΔΠΕ

VOLUME III NUMBER 3 APRIL 1961

Teaching Machines and Self-Instructional Programming ..... Lawrence M. Stolurow and Leonard J. West 1

A Do-It-Yourself Kit for Programmed Instruction ..... Ernst Z. Rothkopf 28

THE DELTA PI EPSILON JOURNAL

A quarterly devoted to research, scholarship and leadership in business education

---

The Journal invites contributions from business educators — members of the fraternity or non-members — from businessmen and women, and from educators in disciplines fundamental to progress in business education. Manuscripts should deal with the theory, practice and research in business education or fields related to it and should be addressed to the Editor.

Published quarterly in the interest of Delta Pi Epsilon. Application to mail at second-class postage rates is pending at Denton, Texas.

© Delta Pi Epsilon, 1960

Extra copies of this issue may be ordered from the business manager. Single copies are $1.00. Orders in lots of 10 will be billed at $.50 per copy. Orders will be filled as long as the supply lasts.

PETER G. HAINES
Editor

WILLIAM HIMSTREET
MARY ELLEN OLIVERIO
LEONARD J. WEST
Associate Editors

EDITORIAL ADDRESS
316 Education,
Michigan State University
East Lansing

BUSINESS ADDRESS AND NATIONAL OFFICE
North Texas State College
Box 6402, NT Station
Denton, Texas

SUBSCRIPTION RATES:
$2.50 per year,
payable in advance

DESIGNED AND PRINTED BY
SEQUOIA PRESS, KALAMAZOO, MICHIGAN

LAWRENCE M. STOLUROW

LEONARD J. WEST

Teaching Machines and Self-Instructional Programming[1]

> It will be a strong testimonial to the weight of tradition over hard empirical fact if "teaching machines" do not receive extensive investigation as possible aids to learning in the classroom.
>
> — WILLIAM K. ESTES
>
> Article on "Learning" in *Encyclopedia of Educational Research*

A larger and larger proportion of an exploding world population is clamoring for more education. At one extreme there are hundreds of millions of illiterates in Asia, Africa, Latin America, and elsewhere for whom education and educational systems must start virtually from a zero point. For the underdeveloped nations of the world education is perhaps more important than economic aid, for eventual self-sufficiency rests necessarily on an educated population, able to cope with the expanding technologies of the modern world. At the other extreme — as in the United States, Soviet Russia, and the western democracies — even highly developed educational systems cannot, under presently typical conditions, keep up with another kind of explosion. And that is, an explosion in the world's knowledge, particularly in science. Furthermore, an expanding science has created an expanding technology. Old jobs disappear and new ones are created. The man who, after an apprenticeship, could practice his trade for life without ever having to learn anything new is a vanishing phenomenon. Today, occupational change is a fact of life; rapid and efficient means of training persons for new occupations are required. In addition, advances in geriatrics, hygiene, and living standards have markedly increased the numbers of older persons; this, by itself, is a significant factor increasing the requirement for education.

An exploding population, the exponential rate at which new knowledge is being created, and the effects on our society of rapid technological change are all factors creating overwhelming pressures for more education. In the face of this need we find an appalling deficit in the production of teachers and classrooms. This deficit promises to become worse, for the production of teachers and classrooms increases arithmetically, while the numbers of people demanding an education and the amount of education to be trans-

[1]This summary article is mainly based on an extensive monograph by Stolurow (1961) and on an article by West (1960). The present article quotes or paraphrases liberally from both without specifically so indicating. It deals with issues judged by the authors to be of interest to the classroom teacher. It is intended as an introduction or orientation to automated instruction and not as an exhaustive treatment. The reader interested in further details should consult the references at the end of this article, particularly Nos. 1 and 4.

mitted increases geometrically. *Training more teachers and building more classrooms cannot ever be a sufficient solution.* Second, there is no way, under present conditions, to cram more education into people short of a substantial increase in the length of the school day, in the length of the school year, or in the number of years of schooling. Third, these needs are world-wide; the requirements of underdeveloped nations are too insistent and too great to await the slow process of large-scale teacher training.

The solution to these problems lies in greater efficiency in the conduct of instruction. Traditional means of coping with these needs cannot ever be sufficient and they are, in fact, not particularly efficient. This is partly because they have tended to treat the symptoms instead of the basic problems (e.g., through manipulating the student-teacher ratio or the length of the school year, or through the development of an imposing array of so-called teaching aids); and it is partly because a number of conditions known to be necessary if optimum learning is to take place simply cannot be adequately met under mass instruction with live teaching.

PROMINENT WEAKNESSES OF TYPICAL CLASSROOM INSTRUCTION

As a basis for suggesting a "New Look" at the process of instruction and for delineating the manner in which teaching machines accomplish their purpose, some major conditions for learning may be briefly examined in the light of conventional instruction.

MOTIVATION AND ATTENTION. One purpose of motivation is to secure the learner's attention. There can be no learning unless the learner is looking and listening. How often can any live teacher guarantee that every student is indeed paying attention?

MAKING THE IMPORTANT CUES PROMINENT. Even if the learner is paying attention, the barrage of words with which instruction is carried out (whether in print in a book or orally from the teacher) often fail to make the important cues sufficiently prominent. The particular thing to be learned is often swamped in the surrounding verbiage. What is required in instructional materials is that they be stripped of excess baggage so as to give adequate prominence to the important cues.

CONTINUAL, ACTIVE RESPONDING. The old phrase, "learning by doing," must be taken seriously. The learner must be busy doing, acting, behaving. In addition to attending, he must be busy making other responses. Typically, however, instruction consists of a teacher talking at students. Much of classroom instruction allows the learner to be passive, let alone inattentive; whereas learning requires active responding by the learner.

IMMEDIATE KNOWLEDGE OF RESULTS. Perhaps the most firmly established (but the most often violated) of all requirements for learning is that of immediate "reinforcement" for responses (in the form of "knowledge of results"). The learner must know, immediately after responding, whether or not his response is correct. The consequences of his behavior must be instantly "fed back" to him. Under typical instructional conditions "feedback" is often delayed (as when test papers are returned the next day or even later) and it is at best occasional rather than regular. This is because

no live teacher can conceivably furnish knowledge of results to more than one learner at one time.

PACING FOR INDIVIDUAL DIFFERENCES. We cannot alter the fact that people differ; nor would we wish to. However, with respect to the differences important for instruction (differences in rate of learning and in the background of information or skill that learners bring to a new learning task) no live teacher can conceivably pace instruction at just the right rate for each learner individually; nor can he ever successfully identify and remedy the variety of weaknesses in background present in any group of students.

INSTRUCTION AS A PROCESS OF COMMUNICATION AND CONTROL

The foregoing basic weaknesses of conventional group instruction suggest the fruitfulness of conceiving of instruction as a *communication and control process* achieved by a system consisting of three elements: teacher, learner, and (instructional) program. Tutorial instruction is the process in which a teacher presents (i.e., communicates) a subject matter to a learner so that he responds to it (i.e., communicates to the teacher) in a way that enables the teacher to determine (i.e., control) the next item of information to be presented. To accomplish this cycle of behaviors, continuous teacher-learner interaction (communication and control) is necessary. The problem becomes one of finding a way in which *continuous* communication and control between the teacher and *every* learner can replace the occasional communication and control characteristic of present mass instruction — without having to provide each learner with a live teacher. Figure 1 is a symbolic representation of a man-machine, closed-loop system for achieving the tutorial condition. In it, the teacher, as a means of *implementing* instruction, is represented by a machine. It reveals that the program as a stimulus affects the learner, whose response is mediated by a machine which, in turn, alters the program — and so on through this cycle.

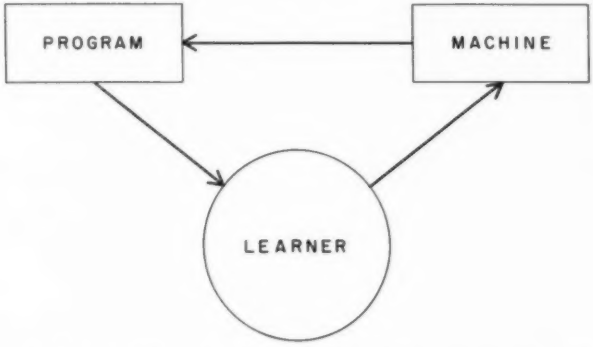


FIGURE 1. Man-machine tutorial system: closed-loop instruction. The machine stands in place of the teacher and takes over the implementation of instruction.

The two basic instructional functions of the teacher are: (a) planning, and (b) implementation. The live teacher, aided by the system conceptualized in Figure 1, will continue to exercise the first function. The second function, however, cannot be efficiently carried out by the live teacher simply because the necessary conditions for efficiency are beyond human capacities under present conditions of mass instruction. Accordingly, the *implementation* of the communication and control process can be turned over to automatic management by properly designed devices with benefit to both teacher and learner.

Certain basic functions must be performed if the necessary conditions for learning are to be met. With instruction conceived as a three-component system of *program-learner-machine*, the design problem is to identify the functions to be met by this system and to maximize the completeness with which these functions are fulfilled.

TEN CRITICAL REQUIREMENTS OF ADAPTIVE INSTRUCTION

It is convenient to speak of ten functions or "behaviors" which must be performed — either by the ideal live teacher or by the ideal man-machine system — in implementing the necessary conditions for adaptive instruction. Further, each function can be carried out in more than one way. Two problems are involved: (a) determining the relative merits of alternative modes of performing the various functions and (b) determining the best combinations to use. However, the functions themselves seem to be critical requirements. These functions and some of their modes are:

1. *Display* function. Presentation of the subject matter in accordance with the nature of the material to be taught (e.g., words, pictures, sounds) is both elementary and essential; but the critical need is to make the cues salient.

2. *Response* function. Provision for a response by the learner may be either for the (a) *selection* type (as in the conventional multiple-choice question in which the learner chooses an answer from a set of options offered), or for the (b) *composition* type (in which the learner composes his own answer, as in filling in a blank in a statement). "Selection" requires recognition whereas "composition" requires recall, and recognition is easier than recall.

3. *Comparator* function. Comparing the learner's response with a predetermined correct response may be done either (a) by the learner, or (b) automatically by the machine. Each response is compared with the standard and a decision about its correctness is reached before the learner makes another response.

4. *Feedback* function. Informing the learner as to the correctness of his response may be either of the (a) yes-no, right-wrong variety, or of the (b) informational and corrective variety (as when the learner is told why he is wrong).

5. *Pacing* function. Pacing refers to control over time intervals between (a) question and appearance of correct answer and between (b) one question and the next. These intervals may either be controlled by the learner (usually variable interval) or by the machine (usually fixed interval). Alternatively, using a mixed arrangement, interval *a* may be learner-deter-

mined, while interval *b* is fixed by machine. (In teaching motor skills calling for high rates of response one might sometimes want fixed inter-stimulus intervals — as a means of forcing the learner to keep up with a predetermined and increasing pace).

6. *Collator-Recorder* function. Making a record of the learner's response to every stimulus (i.e., question) permits: (a) diagnosis of the learner's difficulties, (b) making improvements in the program of instruction, and (c) scoring of performance.

7. *Selection* function. Selection refers to "deciding" what information to present immediately after the learner has responded to the preceding item. The most sophisticated machines "interpret" the learner's responses — whether right or wrong — and feed him the appropriate next bit of information. Secondly, correctly-responded-to items might be "selected out" so as not to reappear.

8. *Library* function. The library is the body of subject matter, the stock of information, which makes up the instructional program and which is stored in the machine. The critical library issues are ones of the storage *capacity* of the device (the amount of information it can hold) and of *access time* (the speed with which the device can pull out an item from its storage for presentation to the learner).

9. *Programming* function. A program is a subject matter organized according to some plan — according to expert judgment or, preferably, according to a set of programming rules. There are two basic types of auto-instructional programs. One is *Linear* or *Non-interpretive* — in which there is no variation in the sequence of information and/or questions presented (although it may be selective, in that correctly-responded-to items do not re-appear). The other is *Branching* or *Interpretive* — in which the learner's response determines the next item to be presented (as when certain types of errors in multiplication send the learner back for further practice in addition). In branch programs, the programmer determines the error possibilities in advance and builds into the program different branches or side sequences to take care of these difficulties. (Linear programming is also called "extrinsic;" branching programming is also called "intrinsic.")

10. *Computer* function. This is a function which, as yet, has been incorporated into only a few of the most sophisticated devices, but which is a prime requirement for a versatile teaching machine system. A computer could have its own program telling it what to do in case of different types of errors. Its function is to modify the program on the basis of the learner's responses. It is the most efficient means of adjusting the instruction to the individual needs of each learner, for it can take into account elaborate patterns of responses in more than one dimension, such as correctness and speed. Furthermore, the computer can process the data obtained.

In connection with the ten functions or requirements just enumerated, it may not be amiss to point out that they simply formalize the necessary "behaviors" of the theoretically ideal human teacher. The human teacher must "display" subject matter at some appropriate rate, must call for responses in an appropriate form, must interpret responses in the light of his own "storage" of relevant information, must furnish feedback to the learner, must select the next information to be presented and/or modify

presentation rate on the basis of the learner's responses, and so on. The question raised is "How can all of these critical functions be performed consistently all of the time, and for each learner individually?" The answer seems apparent. We need a machine, not a man.

FRONT OF CARD

1. Bailment is from a French word meaning "to keep in custody" or "to deliver." Thus, when personal property is placed in the hands of someone other than the owner, we say that a(n) _____ exists.

Ans: _____

BACK OF CARD

bailment

2. You will recall that personal property is property that can be moved; while real property is land and things attached to the land. A bailment does not apply to real property. Only _____ property, which is movable, may be the subject of a bailment.
3. If your family rents the house you live in, no bailment exists because only _____ may be the subject of a bailment.
4. Since a typewriter is personal property, if you lent your typewriter to a friend a(n) _____ would exist.
5. If you lend your typewriter, you expect it to be returned to you. In lending it, you have transferred *possession*, but not ownership of, or title to, the typewriter. In bailments, then, it is not ownership or title that is transferred, but _____.
6. The owner, legally called the *bailor*, is the person who temporarily given up possession of his _____ in a bailment.
7. In the bailment created by lending your typewriter, you, as the owner of the property, would be the _____.

•
•

10. The person to whom _____ of personal property is transferred is legally called the *bailee*.
11. The legal term for the person who receives possession of personal property belonging to someone else is _____.

•
•
•
•

40. Work and services bailments are classed as contract bailments. On the other hand bailments for the sole benefit of one of the parties are known as _____ bailments.

Answers: 1. bailment 2. personal 3. personal property 4. bailment 5. possession 6. personal property 7. bailor 10. possession 11. bailee 40. gratuitous

FIGURE 2. Part of a program in high school business law. (Prepared by Geneva Ewan and Leonard J. West.)

THE PROGRAM

It should be understood, however, that the heart of instruction is the content and organization of the program. The devices (typically hardware) are simply a means of presenting and manipulating an instructional program. In fact, some programs have been prepared for use without hardware. Instead, special sorts of auto-instructional books (see below) are used. The most accurate description of the subject matter here treated is "self-instructional programming" — in which the reference is to any body of instructional materials so constructed and manipulated (with or without devices) as to bring about learning when used independently of the live teacher.

In preparing auto-instructional materials, the subject-matter to be learned is broken down into a series of steps — preferably very small steps, in the present view of most specialists. This is to make cues salient — in a manner and to an extent rarely achieved in the conventional textbook or lecture. The steps in auto-instructional materials are in the form of direct questions to the learner or of a bit of information and then a question. The learner reads the question (or the information and then the question) and either (a) *composes* an answer or (b) *selects* one from a series of options presented. He must respond. Once he does, he is told immediately whether he is right or wrong. Each learner, individually, moves at his own pace through his own copy of the program by the Socratic method of question and answer. Programs and machines differ, however, in the way they determine what happens next.

Some Examples. Some sample items from programs of the composed answer variety are presented first. Figure 2 contains a few items from a program in high school business law on the topic of bailments. With this program all students see the same series of steps regardless of the correctness of their responses.

The bailments program partially reproduced in Figure 2 was presented through what might be called a poor man's teaching machine. That is, each item was placed on a separate 3 x 5 card, with a blank at the bottom of the card for the learner to record his response. The correct response was placed on the reverse side of the card. The process was one of reading the item, writing an answer, and then turning the card over to check the answer against the correct one. For this 40-item program, there were forty cards, and each learner with his own deck worked through the deck, card after card, at his own pace.

Figure 3 displays the first few items in a program in college accounting. These items are in the same form as those of the bailments program in Figure 2. However, the questions are on long sheets of paper (one 42″ sheet contains 38 questions; another 91″ sheet contains 79 questions).

The accounting program goes with a little device that might be called a low-income teaching machine: a stiff paper mask or template with a cut-out displaying one question at a time and a blank space for the answer (see Figure 4). When the learner has written his response to any question, the printed sheet is pulled through the template to the next question. The correct answer to the preceding question is then exposed, together with the next question. The first few "questions" (actually, statements) in the program, shown in Figure 3, illustrate to a nicety the very slow step-by-step fashion in which the items advance and the skillful manner in which the

Assignment I

THE MEANING AND PURPOSE OF ACCOUNTING

Introduction

1. The purchase of a package of gum for five cents is an example of an exchange of _____ for a commodity (the gum).
2. Any exchange of goods or services for money is called a business transaction. The purchase of a package of gum is a business _____.
3. When we pay a 20c Subway fare we are exchanging money for a service. This is an example of a(n) _____ transaction.
4. A $24,000 home is a commodity just as is a package of gum. The acquisition of a new home is an example of a(n) _____.

FIGURE 3. The first four items in a college accounting program. (Prepared by B. Jean Anwyll and Albert E. Hickey of the Itek Corporation, Waltham, Mass., and used by permission of the authors. Copyright 1960.) This program is used with the device shown in Figure 4.

FIGURE 4. A simple paper template or mask for use with a program prepared in the format shown in Figure 3.

very wording of the question virtually gives away the right answer.

However much these programs may look like tests they are primarily teaching devices. Questions are asked in such a way that a wrong answer is hardly conceivable; the prompts to the right answer are plain to see in the very wording of the question. In later items, these prompts are gradually diminished, faded away, or "vanished" — so that the learner has to rely on what he learned from his responses to earlier items in the program. For example, Item 40 in the bailments program displayed in Figure 2 contains no prompts to suggest the answer to the learner; and, as a reflection of the importance of small steps, it is hard to imagine any learner getting Item 40 right without the support of a number of earlier items. With real skill in item writing, as B. F. Skinner has observed, "An understanding of the subject emerges which is often quite surprising in view of the fragmentation required in item building" (1958).

In the preceding examples, the programs require a composed answer. There are other types of composed-answer formats, and there are programs which call for recognition, i.e., multiple-choice, responses. For convenience in exposition several additional types of programs are illustrated in conjunction with the discussion of specific devices, which follows.

TEACHING MACHINES

The steps in any program, or "frames" as they are called, are presented to the learner one at a time. He is not supposed to refer back to earlier frames except as specifically directed and he is not supposed to "cheat" by looking ahead at the answer before making his response. With the 3 x 5 card sequence used in teaching the bailments program and with the cut-out mask used to teach the accounting program, misbehavior on the part of the student is not impossible. With such devices it is necessary to depend on the student's willingness to abide by the rules of the game. Accordingly, and aside from the greater flexibility and versatility of machines over simpler devices, the prevention of misbehavior is one factor supporting the desirability of machine control. Many different machines are in existence; only a few are commercially available now; most, as yet, are for research purposes. Figure 5 shows nine of these machines.

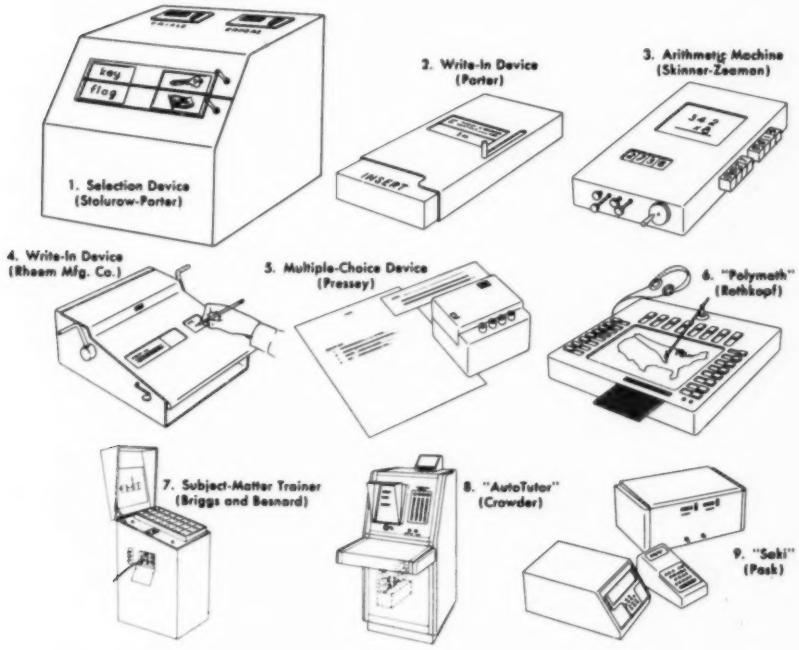


FIGURE 5. Nine teaching machines.

The devices shown in Figure 5 vary widely in size (portable typewriter to kitchen refrigerator), in cost (about $20 to about $4,000), and in sophistication. They vary in the number of functions they perform, in the manner in which they perform their functions, and, to some extent, in the types of learning tasks for which they are intended. Saki (No. 9 in Figure 5) is specifically designed for training in keyboard skills. Six of the other eight devices are designed only for verbal (i.e., symbolic) tasks — which is to say, tasks whose subject matter can be expressed in words, pictures, symbols. This category embraces, of course, a large proportion of all tasks in this world — in and out of school. AutoTutor (No. 8) has a motion picture capability as well as still frame presentation and can therefore be used to teach motor as well as verbal skills.

Of the first eight devices shown in Figure 5, four require a program designed to elicit a selection response (Nos. 1, 5, 7, and 8). Among these, the selection feature, plus the limitation of two choices, makes the Stolurow-Porter device (No. 1) especially appropriate for use with young children. It is designed for paired-associate tasks — teaching the names of objects, a foreign language vocabulary, arithmetic, *et al.* In Pressey's device (No. 5 — a modern Navy model whose antecedents go back to 1925) the student presses a key to select one of four options, and the device keeps a cumulative record. The Subject-Matter Trainer (No. 7), developed by Briggs and Besnard for the Air Force, accommodates 20 response options and can be used under "prompting" conditions (correct answer shown to learner before he responds), under "confirmation" conditions (correct answer shown after response), or under test conditions (no knowledge of results). The AutoTutor (No. 8), developed by Crowder for the Western Design Division of U. S. Industries, is distinguished by a very large library or storage capacity (up to 10,000 items or frames), access to any frame in the library at any time (24 frames per second), automatic recording of time plus step sequences selected by the learner, and, an interpretive (branch) program. How does it work? It is a daylight microfilm viewer with a panel of push buttons to accommodate the learner's responses. He reads the material on the viewing screen, decides what his answer is, and punches into the response panel the frame address associated with that answer. He then pushes the "view" button or the "motion" button, whichever is appropriate. Once he does this the projector goes to the address and displays the content. The new material tells the student whether he is right or wrong and what to do next. Thus, he selects, and the machine corrects or reinforces, as appropriate, while also moving the student backward or forward through the program as appropriate.

Four other devices among the first eight (Nos. 2, 3, 4, and 6) require the learner to *compose* his own response. In the Porter and in the Rheem devices (Nos. 2 and 4) the learner writes in a response (typically a word or words which fill in a blank in a statement — as with the bailments and accounting programs displayed earlier in Figures 2 and 3).

The sequence of events used in write-in machines in general may be illustrated by the behavior of device No. 4. The question or frame appears in the rectangular opening at the left. The learner writes in his answer at the right. Depressing the lever at the left front of the machine raises the shutter over the square area at the center, exposing the correct answer to the student's view and, at the same time, putting his composed answer under

glass so that it cannot be tampered with. The handles toward the rear of the machine accommodate left- and right-handed persons and are used to bring a new question into the display window.

In the Arithmetic Machine (No. 3) the answer is composed by setting plungers. A new problem will not appear in the display window until the preceding problem is answered correctly. "Polymath" (No. 6) is potentially quite a flexible device. It could accommodate a number of different types of materials, e.g., drawings, geometric figures. As shown in Figure 5, it is used to teach the selection of the proper route on a map. A printed circuit under the display gives the learner immediate knowledge of results as he moves his stylus.

All the devices shown in Figure 5 have in common *immediate reinforcement* — the furnishing of immediate knowledge of results, of immediate feedback to the learner. In one way or another (by buttons, switches, levers, handles) each of the devices is designed to follow the learner's response with a display of the correct answer and with a new question — or else with a refusal to budge, signaling that the learner's response is wrong.

All the devices thus far discussed *as devices*: (a) display stimulus materials, (b) call for an active response to each stimulus item or frame, (c) furnish immediate feedback, (d) allow the learner to pace the task, and (except for AutoTutor) (e) accommodate linear programs, and (f) have small storage.

Saki (No. 9) is designed to train key-punch operators for data-processing installations. It is "tailor-made" for instruction in typewriting. In fact, Gordon Pask, its inventor, has reported work in process on an adaptation of Saki for teaching typewriting (see Lumsdaine and Glaser, 1960, p. 338). Saki both teaches and learns. It issues commands to the learner or it obeys the learner's "commands," as the situation may demand. In teaching typewriting, for example, if the learner can keep up with the rate of presentation of copy materials, Saki "knows" it and will increase its rate of stimulus presentation so as to "lead" the learner. If the learner falls behind, Saki "knows" that also and slows down its rate of stimulus presentation in keeping with the learner's proficiency. As E. Z. Rothkopf has put it (see Lumsdaine and Glaser, 1960, pp. 319-320):

> The simple Saki computer plays a game with his student which few seem able to resist. Saki races his pupil and the pupil's attempts to catch up make Saki run even faster. Mr. Pask reports interesting elation effects in connection with these races. Students are said to underestimate the amount of drill time which has passed and, further, some students are reported to practice until they are exhausted . . . A most interesting research issue here is how far the carrot should be kept from the donkey's nose in order to make him run the fastest for the longest time . . . the relationships between performance, pacing, and learning is at this time one of the most challenging empirical questions for the automation of sensory-motor skills training.

How much more can one wish in a typewriting teacher, and what human teacher can come within sniffing distance of Saki's matchless responsiveness to the learner's behavior?

MACHINE ADAPTIVITY

"Adaptivity" refers to the capacity of a machine and its program to adapt, in one or more ways, to the specific needs of the individual learner and is thus the most critical feature of automated devices. The more functions performed by the machine the more adaptive it is. Two *minimally adaptive machines* are the card deck and the cut-out mask described in connection with the bailments and accounting programs, respectively. Basically, they display information, require a response, and provide feedback or knowledge of results for every response. Programming is for the most part linear — errors are not interpreted and the sources of the learner's weakness are neither identified nor worked on. *Partially adaptive* machines perform more, but not necessarily different, functions and they exercise greater control over those functions than do the minimally adaptive machines. One prominent new function is manipulation of the order of events on the basis of the learner's errors — in one manner or another, incorrectly-responded-to items reappear. The machine in the upper left corner and the arithmetic machine (Figure 5) are examples of partially adaptive machines. The "correction" procedure is built into them. AutoTutor, (No. 8 in Figure 5) is a partially adaptive machine which has an extensive library and random access to items in it. This makes possible interpretive programming. However, all of its "decisions" are based only upon the most recent response and its correctness. It records decision time as well as responses, but does not process the data.

The key feature of the *fully adaptive* machine is a computer for: (a) the generation of program steps; (b) automatic selection of stimulus materials (and/or control over their rate of presentation) according to sets or patterns of learner's responses; and (c) data processing. These functions are possible because the computer senses, keeps track of, and processes masses of responses—not just the immediately preceding response. Saki, (No. 9 in Figure 5) is such a device. PLATO (Programmed logic for automatic teaching operations) developed by Bitzer and Braunfeld at the Coordinated Science Laboratory, University of Illinois, is another such device (not shown in Figure 5) and one that is more versatile than Saki in terms of the varieties of subject matter it can accommodate. Furthermore, PLATO can present program steps that require the learner to compose his response but which, nevertheless, also are of the interpretive or branch type. Unlike less adaptive machines, it can perform the comparator function for the learner even though his response is constructed.

In general, the fully adaptive machines perform automatically all ten of the critical requirements outlined earlier.

PROGRAMMED AND SCRAMBLED TEXTBOOKS

Another means of implementing the major requirements for adaptive instruction is through the use of new printed formats which require no hardware. Each is specially designed to accomplish several of the functions built into the machines previously discussed. A number of these specially designed auto-instructional books are described next.

PROGRAMMED TEXTBOOK

The simplest in appearance and format is the "Programmed Textbook,"

which generally calls for the composition of answers by the learner. A few sample pages from a primary-grades arithmetic program are illustrated in Figure 6.

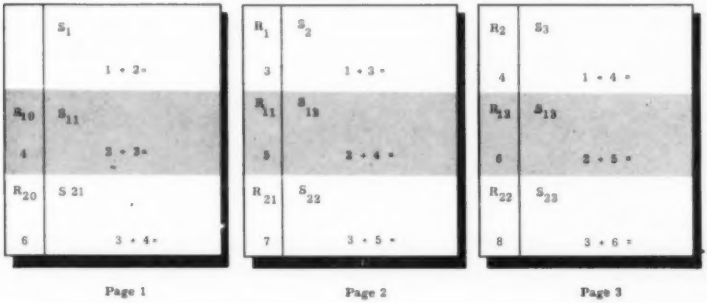


FIGURE 6. An illustration of 3 successive pages in an arithmetic *programmed textbook*.

As illustrated in the upper panel of page 1 in Figure 6 a problem (S_1, that is, Stimulus 1) is given. The learner writes his answer in the space provided. Then, turning to page 2, he finds (at R_1, that is, Response 1) the correct answer to Stimulus 1 and a new problem (S_2). To make more efficient use of space the learner works through a chapter using only the top row of panels. Then he goes back to page 1 and works through it using the next row of panels, etc. The number of panels on a page is uniform and depends on the space requirements for items. This, of course, depends, in turn, on the nature of the subject matter taught.

In the format shown in Figure 6 the answer to the previous question and the new question appear on the same page. Another common format (not shown here) puts each answer on a page by itself. This doubles the page requirements but has the possible advantage (over the format shown in Figure 6) of avoiding the distraction of having an old answer compete for visual attention with a new question.

With either of the two formats just described, questions or problems may be preceded by information which furnishes the learner with a basis for answering the question presented. This feature is not illustrated here (and is, in fact, not necessary in the particular instance of this part of an arithmetic program).

Two things can be noted about the *programmed* textbook — in contradistinction to the *scrambled* book described next. In the programmed textbook, (a) feedback is immediate (next page), and (b) the program is noninterpretive — all learners go through the same materials regardless of whatever errors may be made — their responses do not determine what they see next. Usually, but not always, answers are composed in program-

med texts; and this mode of response with a printed format usually requires a linear program.

Cutback Page Book. There are other variants of the programmed textbook form of self-instructional programs. One is described here in detail. It was developed and first used by Stolurow at the University of Illinois. It is the "cutback page book" with separate answer sheet, and is illustrated in Figure 7.

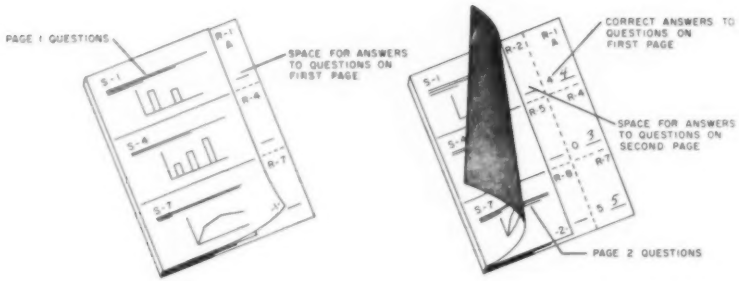

PAGE 1 QUESTIONS
SPACE FOR ANSWERS TO QUESTIONS ON FIRST PAGE
CORRECT ANSWERS TO QUESTIONS ON FIRST PAGE
SPACE FOR ANSWERS TO QUESTIONS ON SECOND PAGE
PAGE 2 QUESTIONS

FIGURE 7. Cut-back booklet with separate answer sheet.

Successive assemblies (e.g., sets of five pages, in which each page in any set is narrower than the preceding one) economize on paper and thus on production costs. Furthermore, a separate answer sheet fits under each set of pages and makes the program reusable, while also furnishing the learner and the teacher with a convenient and complete record of responses. A crucial feature is that the answer sheet is so arranged that when appropriately inserted behind the last page in each assembly, or set, feedback is as immediate as in the programmed textbook described above and markedly faster than in the scrambled textbook, described next. A column for writing answers extends beyond the edge of each page, whereas the corresponding correct answers are covered by the text page and are only revealed to the learner, one by one, for comparison purposes as he turns the page after recording his own answer. In other words, the cutback page book implements the features of the programmed textbook, redresses two weaknesses in the scrambled textbook (delayed feedback, no recorded response) and does so at a dollar savings estimated at about 25 per cent.

SCRAMBLED TEXTBOOK

In the scrambled textbook (as illustrated in Figure 8), responses are of the recognition type (i.e., multiple choice), and the program is interpretive — that is, the learner's response determines what he sees next.

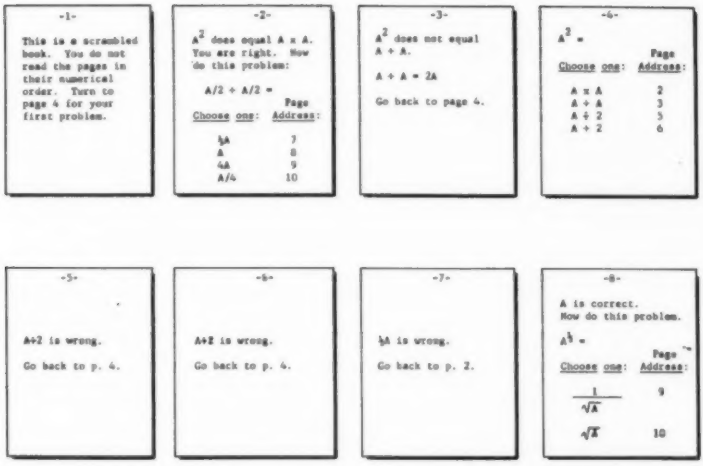


FIGURE 8. A set of pages illustrating the Crowder scrambled text. (Note that space does not allow for an explanation which could otherwise be included to tell the learner why he is wrong.)

In a scrambled textbook the pages are bound in serial order, but the learner reads these pages in an order that is scrambled. The reader starts on page 1 (see Figure 8) and is sent to a page other than number 2, e.g., 4. If, in reading page 4 he thinks that A^2 means or is equal to A + A, he is "addressed" (i.e., directed) to page 3, on which he finds some information aimed at "unscrambling" his erroneous notions and is sent back to page 4 for another try. When he makes a correct choice of answer (as in being addressed to page 2), he is told, "You are right" and is given the next bit of information and an associated new question. Ordinarily questions are preceded by a few sentences of information leading to the question. Similarly, wrong answers are often accompanied by explanation or clarification. For example, in another scrambled book on computer number systems, the learner who answers "9" to the question: "What product will be reached by using 2 as a factor three times?" is told: "You're showing the right spirit, but you got the problem backwards and used 3 as a factor twice . . . we want you to use 2 as a factor three times . . . Now return to page 5 and try again." Explanations of this sort are typical but are not included in the sample shown in Figure 8 because of space limitations.

Norman Crowder, who developed the basic concepts and the first examples underlying scrambled textbooks — his versions are published as TutorTexts — has incorporated the scrambling notion into film for use in an AutoTutor (No. 8 in Figure 5), described previously. Mainly, the machine simply does at high speed (24 frames per second) what the reader of a printed scrambled textbook does manually at a slower rate when he turns

pages to locate the "address" corresponding to his choice among the alternatives. This may be a very important advantage since the immediacy of feedback is so important. Another advantage of AutoTutor over *TutorText*, its textbook version, is that the machine also can project motion picture sequences to display motion and information about motor skills. A third merit is that the AutoTutor has a unit which automatically records each response in the order made and the elapsed time between responses, thus permitting scoring of performance and identification of trouble spots in the program. There is no record of the student's responses when he uses a scrambled book.

Although self-instructional programs in textbook form do not make cheating impossible, it has not been a teaching problem to date. The programs have been so skillfully constructed that there is little if any need to cheat. Furthermore, the cheating problem is, to all practical purposes, nonexistent with the scrambled textbook since the learner does not know where to go next until he commits himself to one of the available alternative answers.

SOME RELEVANT ISSUES

The "Prominent Weaknesses of Typical Classroom Instruction," discussed earlier, may now be contrasted with conditions under auto-instructional programming: (a) a question to be answered or a blank to fill in are as attention compelling and as difficult to ignore as a ringing telephone; (b) the requirement of overtly answering questions keeps the learner continually making active responses; (c) since there is a device and a program for each learner, instruction is tutorial; (d) in most cases each learner works at his own best rate; and (e) each correct response is reinforced. Further, (f) the small steps and careful sequencing of items in these programs tend to preclude leaving any learner behind because of gaps in his knowledge; (g) the stripped-down language, the brevity of each frame, and the deliberate use of various cueing techniques (e.g., underscoring of key words) focus the learner's attention on the important cues; (h) for each learner every response is immediately followed by knowledge of results.

Taken together, teaching machines and self-instructional programming indeed represent (a) an application to the instructional process of modern learning theory and current research; and (b) the application of modern technology to fulfill the necessary conditions for learning. The result is tutorial instruction using the Socratic method, carried out with a level of control quite impossible under standard live instruction.

There are, to be sure, a number of issues raised by this new concept in education to which it is worth calling attention. Perhaps the most prominent among these is the question:

DO TEACHING MACHINES DO AWAY WITH THE LIVE TEACHER?

The answer is "No!" As indicated earlier, someone has to plan the materials or programs. Secondly, some subject matters (or parts of them) are intrinsically susceptible to programming while others are not. Third, the extent to which program-and-machine perform all the necessary functions of instruction — the sophistication or adaptivity of program and machine — determines the extent to which a live teacher also must work with the students.

WHAT KINDS OF LEARNING TASKS CAN BE PROGRAMMED?

The primary requirement for programmability is that there be agreement as to the truth of any given set of statements describing the content of the field. To any set of questions incorporated into a self-instructional program there has to be agreement on what are the right answers, or there has to be data on the percentage of persons in identifiable groups who select each response option. For example, we can teach grammar, spelling, punctuation, parts of speech, and sentence structure through this new medium — but we do not expect to teach creative writing. There is no one right answer to what is or is not a "stylish" piece of writing. However, we might approach the problem of teaching the child to be analytical — to identify poor writing, and to evaluate the writing of others — all of which could indirectly contribute to his own creative writing. Likewise, we could teach someone through this new medium all about the election process and the mechanics of voting; but we would not presume to incorporate into a teaching program a question on the order of: "Among the presidential candidates in the American elections of 1960 the person to vote for is ....." The same is true for other issues and problems — for example, ethical issues — which demand the give and take of creative (and informed) discussion and to which there are few, if any, canned answers.

There are two other prominent restrictions or criteria for tasks that can be programmed. One is governed by the form and/or the extent of the responses called for by the task. Composed responses which exceed more than a few words are not readily susceptible to *rapid* and *reliable* comparison automatically by a machine (e.g., PLATO) or by the learner himself, as when a model answer is shown him (e.g., AutoTutor). Furthermore, as responses become more complex and long the comparison with the standard is slow and knowledge of results is delayed thereby. Another criterion of programmability (branching type) with selection responses (e.g., AutoTutor) is that the programmer must be able to foresee one or more plausible wrong answers. Failure to foresee and to deal with a popular misconception can leave the learner with a wrong notion that may not be dislodged by the remainder of the program. Both these criteria relate to the adaptivity (see below) of a teaching machine system.

In any event, the present scope of self-instructional media may be inferred from some of the learning tasks which have already been programmed: accounting, algebra, arithmetic, binary arithmetic, Bridge, card punching, chemistry, chess, educational psychology, (fundamentals of) electricity, electronics, elementary number theory, geography, German, Hebrew, history, law, machine operation, map reading, (fundamentals of) music, navigation, philosophy, physics, psychology, radar operation, reading, Russian, Spanish, spelling, statistics, trigonometry, *et al.*

Adaptivity of Program and Machine. Assuming that the subject matter to be taught is one in which there is agreement among experts as to what are right answers to relevant questions, a number of other requirements must also be met.

For example, knowledge of results immediately after every response is essential in order to provide *Feedback*, thus fulfilling the pre-eminent condition of *reinforcement* for correct responses. For the machine to perform this indispensable function the learner's response must be in a form which either the device or the learner can readily interpret. Thus, a standard must

be available with which the student's response can be compared. For example, a device will "speak" good German to the student of German, but it cannot check the precision of the student's own pronunciation. But it can play back his speech so that he hears it along with the model and can make the comparison himself — the student thus providing the *Comparator* function. Machines differ in the extent to which they (can) automate this as well as other functions.

The most critical requirement for a maximally adaptive machine (i.e., for truly individualized instruction) is the need to adjust in one or more ways to the specific needs of the individual learner. When the learner makes a mistake, what should happen next? The various possibilities are illustrated in Figure 9, in which the basic psychological elements are represented as a time series going from left to right. The four elements in this series are: (a) the cue-stimulus or question, S_i; (b) the response, R_i; (c) knowledge of results, KR; and (d) the sequential action of the program — its forward or backward movement as mediated by the *Selector* unit.

When the learner makes a mistake, if the program is a linear or non-interpretive one, the items are in fixed and rigid order. The only course of action is for the learner to be shown the right answer and to be moved ahead to the next item in the program, as illustrated in the top section of Figure 9.

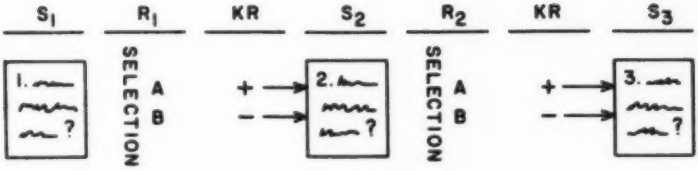
Notice (top line, Figure 9) that both correct (+) and incorrect (—) responses (immediately followed by appropriate knowledge of results) lead to the same next step. That is why it is called a linear or non-interpretive program.

Moving the learner ahead regardless of his behavior is not a very clever tactic, and we would much prefer to react differentially to right and wrong responses. To do this we need to know why he went wrong. This is where branching, or interpretive, programs enter. A first but significant step in this direction is the *correction* procedure, illustrated in the second section of Figure 9. The critical difference between this circular or error-loop program and a linear program lies in the differential sequential action of the program. In both instances, the right response takes the learner to the next step. In the error-loop program, however, a wrong response returns the learner to the original question and requires him to make another choice of answer. He moves ahead only when correct — so that his last response to any stimulus is always a correct response.

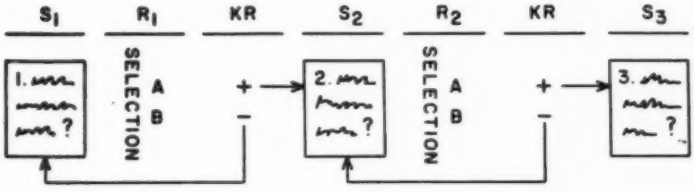
Two more sophisticated forms of branching are shown in Figure 9 (bottom two sections): (c) condensing branches, and (d) expanding branches. (Many variations are possible, and the ones shown are simple ones.) The first of the two shown is also called "forward branching," but the label, "condensing" more pointedly conveys its effect upon the length of the program as experienced by the learner. With it, a right response skips the learner ahead by one or more steps. A wrong response moves the learner to the next step.

The expanding branch is the basic form of branching and is illustrated at the bottom of Figure 9. With it, the branch is built into the error loop. A wrong response takes the learner to a supplemental sequence in which he receives a help or a prompt and then returns to the original question.

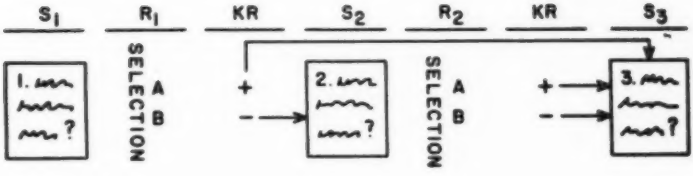
It should be noted that these are basic forms which can be used as such or in various combinations (e.g., the circular or error loop along with the



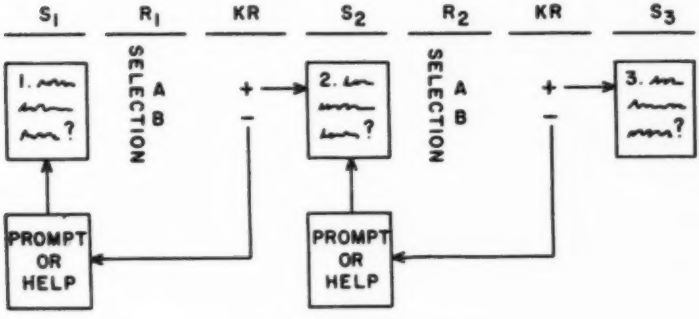
(a) Linear or non-interpretive



(b) Circular or error-loop program using correction procedure



(c) Condensing branch or "forward branch" program



(d) Expanding branch program

FIGURE 9. Symbolic representation of four types of programming sequences.

condensing branch form). Neither the linear form nor the condensing branch form utilizes the correction procedure, and they are therefore minimally adaptive. The error loop forms and its variants using the correction procedure add an important dimension to adaptive programming.

The programmer's first task is to make the learner an active participant in the teaching-learning process. If the teaching system is to be adaptive, it has to take individual differences (in aptitude or knowledge) into account. This is what makes live teaching difficult: it is easier to "tell" someone than to ask questions and to correct answers in a systematic manner. The latter requires the teacher to be a diagnostician as well as a source of information. In any event, the development of a maximally adaptive teaching program requires considerable skill in anticipating error possibilities and in building appropriate branches or side sequences into the program.

TEACHING MACHINES — TEACHER OR TEACHER'S AID?

With a maximally adaptive program and device — in which a branching program interprets the learner's responses, provides immediate knowledge of results, and feeds him the appropriate next bit of information depending on his preceding behavior — instruction is wholly self-contained. In portions of courses where this can be accomplished there is no need for live teaching. Thus, in different ways for different courses, the teacher's role will be changed. These new devices could aid in a significant way and free the teacher for other things.

Two additional observations may be made. One is that teaching machines — unlike the so-called teacher's aids or audio-visual aids (e.g., books, movies, film strips, TV) — are based upon a different rationale, one that is much more sophisticated. These more familiar devices can never be more than aids since they merely present stimuli — they start and stop with just the first step in instruction. They are one-way communication systems. Necessarily, some other agent (the teacher) must be available to perform the other critical requirements for efficient learning (e.g., reinforcement, correction). The typical so-called "aids" communicate but they do not control behavior; that is, they do not elicit a response from the learner. Consequently, they are wholly unresponsive to the learner's behavior. With teaching machines, responses are required and are followed immediately by knowledge of results. As machines perform more of the ten critical functions (described earlier), their adaptivity increases and they are, correspondingly, more and more a teacher and less and less an aid.

A second observation is that, unlike the procedure with most conventional aids, in using a teaching machine the teacher does not do all over again what the machine-and-program do. Instead, the live teacher continues where the machine-and-program leave off. They do not do the same things differently; rather they do different things. Commonly, the student is assigned a chapter in his textbook and comes into class the next day to hear his teacher expound all over again the material in the text — or the reverse procedure (of hearing the teacher expound and then going home to read the text) is used. This redundancy is a consequence of the inherent difficulties of typical mass instruction. With a self-instructional program the live teacher can take up where the program stops. In fact, the reported experience of teachers involved in self-instructional programs has often been one of elation and astonishment at the fact that they do not have to spend their

time grinding away at the groundwork of their subject but are indeed free to deal with more advanced or complex issues: their students come to them with the groundwork already laid.

WHO WRITES PROGRAMS?

Does each teacher write his own program? Ordinarily, no![2] Writing a good program requires a combination of at least four talents not often found in one individual: (a) expert mastery of subject matter, (b) skill with words, (c) a firm understanding of the psychology of learning, and (d) facility with the processes of program construction and evaluation. Program development, therefore, is often a team enterprise at the present time. It is hoped, however, that psychological research will provide us with principles for programming which teachers could use in building their own programs.

With respect to techniques for programming — the ground rules and mechanics of writing and sequencing steps — these are accumulating at a rapid rate as researchers acquire more experience (see Stolurow, 1961, Chap. VI. for details). However, writing steps for instructional programs is to some extent an art, and some people are better at it than others. At the same time, while programming techniques are, at present, little more than a collection of rules-of-thumb, some of them can be stated in terms which make them susceptible to experimental verification. Accordingly, one can look forward to progressively more science and less art in programming. This is, of course, all to the good, mainly because better programs will result and partly because more people will be able to program. Stated very broadly, programming requires (a) identifying the responses that constitute the criterion behavior (both immediate and long term) — that is, the responses representing achievement of the objectives of instruction, (b) identifying the stimuli to which the responses are to be associated, and (c) organizing the stimulus-response terms into particular sequences of items.

COSTS OF AUTOMATED INSTRUCTION

Programming is extraordinarily time consuming — rather more so than writing an ordinary textbook. There are several specific reasons for this, but mainly it is the amount of detail required in programming that accounts for it. For example, the programmer must work at the level of simple subject-predicate relationships. After building one such relationship, other associations have to be built on it so that larger and larger sets of associations are formed. Programming is analogous to placing individual bricks together to construct a building; the program steps are like the bricks. The program is designed to teach: to build knowledge or skill, to build associative connections into larger and larger sets — a conceptual structure.

Another reason programming is time consuming is that it is necessary to explore different sequences of steps to see what responses the learner will make. This is time consuming because the optimum program often becomes apparent only after several sequences have been written and tried out. Once the alternatives have been tried out the programmer must select the one which he feels achieves his objectives best. Since programs differ from texts in having a "built-in" basis for evaluation, they typically are given to an

[2]For suggestions on how teachers can perform as programmers, see Dr. Rothkopf's article, immediately following this one.

appropriate group and the responses recorded. For linear programs the error rate is determined. With linear programs high error rates mean that the program needs rewriting. In branch programs the error rate is not minimized; rather the distribution of errors over alternatives is important, for the different types of errors have implications for branching. With a textbook the evaluation is more difficult and much less exact. Consequently, even if the author asked people to read chapters and comment on them he would not obtain data of sufficient precision to permit detailed revision. In writing programs, on the other hand, items would be revised, discarded or relocated with a patient attention to the smallest details that the textbook writer never dreams of and could not accomplish even if he wished.

The amount of time required to write multiple choice and constructed response steps may differ, and it is, in any case, considerable. Rigney and Frey recently summarized estimates of writing time (including all tryouts); they varied from 2 to 16 per day of finished steps for constructed responses and from 8 to 10 per day of the multiple choice (branch) type. Most programs run to at least many hundreds of items and many of them involve several thousands of items. The expense of developing an effective program for a year course is formidable. This means that programmed learning is economically feasible in public education only for subject matters taught to many thousands of persons or for behaviors which are very costly to teach and very important anyway (e.g., medical diagnosis).

One important cost factor is the development of a durable device. Some cheaper devices now on the market (designed for printed programs) cost $20, but they were not tried out and pretested. Microfilm, however, is said to be substantially cheaper to use than printed materials since many learners can reuse it. Often, the printed programs used in devices are consumed by each student. Thus, the machine cost is not the whole story. Programs on paper more readily permit insertions and changes — something the schools can often accomplish themselves (by means of inexpensive duplication processes).

In any event, while the cost of program preparation is high, and the outlay for machines is not negligible, the question is one of whether these costs will be more than balanced by increased learning in less time and by reducing the extent of the need for additional teachers. Of course, it is the development of a new program that is especially expensive. Once programs are developed the cost of revisions should not be large.

REFINEMENT OF PROGRAMS

The earlier mention of a tryout of steps in a program deserves special emphasis because it contains part of the reason why programmed learning might be superior to conventional instruction. When the first version of a program is inserted into an appropriate device, not only is the learner furnished with feedback for his responses, but the records of his responses furnish the programmers with precise feedback as to the quality of the instructional program. Steps which are too difficult are readily identified, and new items can be composed and inserted into the program. Similarly with other kinds of deficiencies. For example, in the original version of the bailments program shown in Figure 2, "possession" in the second sentence of Item 5 was not underscored, and one-third of the learners got the item wrong (responding with "property" or with "your typewriter"). When

"possession" was underscored (thus giving prominence to the relevant cue), the percentage of correct responses rose from 66 to 96. Contrast this with typical classroom instruction in which it is very difficult to determine with any real precision what the students' difficulties are. One investigator has rightly remarked that these programs "teach the teacher how to teach."

Besides serving as a basis for revision of programs, the individual's record of responses identifies his weaknesses and allows the teacher (or the student himself) to work on particular weaknesses and difficulties. Thus, individual educational diagnosis not only is possible but also can be quite precise.

THE "HUMAN ELEMENT" IN TEACHING

On an emotional level, persons with adverse reactions to the notion of teaching by machine use such words as "cold" and plead for human "warmth" as an aid to learning. While recognizing the prevalence of objections of this order, it may be suggested that the learning process is one requiring rather more light and rather less heat. We do not yet know whether automated instruction can achieve all the things people have pointed out, let alone have "inspirational" qualities like those so commonly ascribed to the live teacher. An automated device cannot raise its voice, shrug its shoulders, or indulge in the other little shadings and nuances of which the human teacher is capable. But let us not delude ourselves about live teaching skill. Teaching skill is normally distributed in this world: a small proportion of teachers are marvels to behold; an equally small proportion are a perfect disgrace; most teachers fall somewhere between these two extremes.

As John W. Blyth has pointed out, words like "machine" and "automation" are emotion laden. "Thus the phrase 'automation of education with teaching machines' represents such a summation of horrors for some people that it blocks intelligent inquiry into the merits of teaching machines . . . In the end, the widespread use of teaching machines will not depend on emotional reactions to a word but on the machines' value in helping us cope with urgent educational problems" (see Lumsdaine and Glaser, 1960, p. 401).

To the objection that learning proceeds best under social conditions, that learners often profit from the behavior of other learners, it may be suggested that no group ever composed a symphony, created a scientific theory, or painted a picture. Besides, the "social" behavior of learners in a group can often be a source of distraction which impedes rather than aids learning. Might it not be better for the student to learn alone and, where appropriate, to use what he has learned in a group setting? This question can be raised now with a reasonable expectation of being answered, whereas this was not possible before the teaching machine.

IMPLICATIONS FOR THE ORGANIZATION OF INSTRUCTION

Under present conditions a "C" grade means that a student has mastered about three-quarters of the test questions. Under programmed instruction, in which everyone proceeds at his own rate — assuming that the task is not beyond the capacity of the learner and that the program is a good one — all may master the course at a higher level but in greater or lesser time, as the case may be. The slow student takes the time he needs; the quick one,

in the meantime, romps along through his various programs at a rapid rate. Time required to finish the program may become the motivating device. It is hard to imagine more powerful motivation for attending to business than a payoff in time.

Consider, also, the effects of presently typical conditions on the student who is not particularly bright. He sits in a class with others, making public responses, and his inadequacies are continually being advertised to his classmates. Some children come to hate school and to resist all learning not only because they do not learn, but also because their failures are publicly advertised. Under programmed instruction, performance is a private matter between learner and teacher. Some argue that auto-instruction is even more democratic than live teaching since all students are treated alike — rich or poor, homely or attractive, white or negro, Christian or Jew — and all are evaluated on a basis free of such irrelevancies.

In fact, with self-instructional programming used, say, in a third-grade class, Johnny Bright might be working at his program in sixth-grade arithmetic, while Tommy Dull, his neighbor, might be working on his program of second-grade arithmetic. Susie Average, on the other hand, has finished her daily dose of third-grade arithmetic and is working on her program in spelling. The adult in this room at this time may not be the teacher, but a technical aide. Perhaps he has an oil can and a screwdriver to deal with minor mechanical troubles. But mainly his duties are to collect the response records from programs which children have completed and to insert new programs into the devices. Or perhaps he supervises the movement of a child to and from the machines. Sensibly spaced sessions at machines are interspersed with activities that do require the live teacher, who takes up where the machine leaves off to teach what the machine has not taught or cannot teach, to inspire, to delight, to amuse, to challenge.

Carry this illustration to its logical conclusion and we find that for many courses the question of homogeneous versus heterogeneous grouping of learners tends to disappear. We might wish to group students homogeneously for social purposes (e.g., on the basis of chronological age). But this same group is working at a number of different tasks and at a wide range of levels within tasks. This is, of course, the ideal: each person working to his fullest capacities and at whatever rate and on whatever tasks his capacities permit. The point is that teaching machines can make this accomplishable without making impossible demands on the teacher.

AUTOMATED INSTRUCTION — CURRENT STATUS AND CURRENT ISSUES

Self instructional programming and teaching machines are so new that it is premature to speak of definitive findings, of having firm answers to the innumerable questions that this new development engenders. Present findings are provocative and tend in the directions indicated in the statements which follow.

1. Savings in instructional time are characteristic (e.g., concepts taught to children in 15 minutes that had taken 65 minutes under live instruction; spelling and vocabulary taught in one-fourth to one-third the time taken under live teaching).

2. Beneficial effects do not seem to wear off. Novelty, and the usual

beneficial effects of novelty, seem *not* to be a prominent source of superiority. Performance scores for the second half of a year were the same as those for the first half of that year — showing that novelty effects, if any, did not wear off in the course of an entire year of instruction.

3. Programmed instruction reduces the variability of scores. Individual differences are reduced and those of lower ability become more like higher ability students.

4. Aptitude test scores tend to lose their predictive validity when more efficient methods of instruction are employed. In other words, the moderately high correlation between general ability measures (like I.Q.) and scores on learning tasks seems to drop to zero when the learning materials are programmed for most efficient learning.

5. Students typically (but not uniformly) like machine instruction and prefer it to live teaching. Automated instruction seems to be self-motivating.

6. With thoroughly pre-tested and appropriately revised programs, superior test scores have been achieved under automated instruction over those achieved under live teaching conditions.

7. Teaching machines have successfully taught appropriate tasks to persons ranging in ability from mental retardates to gifted graduate students.

SOME CURRENT ISSUES

Some of the issues raised by automated instruction can at present be stated only very generally; others have a sufficient background to permit more detailed specification, as follows:

1. What is the role of automated instruction in a total educational scheme? For what purposes can it be the sole source of instruction; for what purposes will it be an adjunct to live instruction? (We have yet to explore the full range of complexity of behaviors potentially susceptible to automated instruction.)

2. What is the learner like? Is he a receptive mechanism (*tabula rasa*) who simply forms associative connections that mirror his experience? Or is he a selective mechanism who selects and extracts information from the environment? The former assumption calls for a (composed-answer) program that minimizes errors by the learner; the latter assumption utilizes errors (as in multiple-choice programs) to build knowledge and skill. Possibly both apply, but at different stages of development.

3. What kinds of responses are critical and most efficient? Should some programs be designed deliberately to build responses by requiring them to be covert and others (probably later on) require overt responses and build up speed? When and how can we use constructed (recall) responses and multiple-choice (recognition) responses most efficiently? (Recall scores increase rather regularly with age whereas recognition scores do not show regular increments with age. Recognition is easier than recall; accordingly, programs for young children will be more readily mastered if they call for recognition-type responses.)

4. Much (but not all) of the evidence suggests that general intelligence test scores are positively correlated with initial and final performance scores on a learning task. They predict status or achievement. However, they have virtually zero relationship with *gain* scores, or rate of change. Thus, an I.Q. score is a measure of the learner's capacity for a task rather than a predictor of the rate at which he will learn what he is capable of mastering. Individual differences in rate of learning are related to specific aptitudes, differences in learners' immediate memory span, attention habits, motivation, and to specifics of the particular learning situation.

a. Thus, it is not efficient to prepare different programs for learners who are at different levels of general ability. All those above a certain minimum of general ability can learn from the same program. But since specific aptitudes may correlate with gains on a particular learning task, future research might reveal that aptitude scores can be used as a basis for differential programming. Consequently, determining the particular aptitudes required in learning different parts of a program is an important problem for future research.

b. Defining the ability minimums for particular tasks is another important research problem. Time should not be fruitlessly spent attempting to make a silk purse out of a sow's ear.

5. Continued exploration of interactions between learner variables and teaching-method variables is needed.

6. Experimental verification of rules for programming is needed. Much programming has been carried out on the basis of *ad hoc* rules of thumb—some of which have substantial basis in earlier findings, others of which presently represent best guesses.

The preceding six points are merely a few of the prominent issues, and under each an army of sub-issues and sub-questions is subsumed.

TEACHING MACHINES — FAD OR FIXTURE?

Whether teaching machines are fad or fixture is a question the answer to which is at present necessarily a matter of opinion. The facts to settle the question are probably several years in the offing. A dig-in-the-heels resistance to change is hardly new, and Sidney L. Pressey as early as 1926 complained about the cultural barrier prohibiting the application of modern technology to the teacher's problems. More than thirty years later B. F. Skinner found it necessary to point out that: "In any other field a demand for increased production would have led at once to the invention of labor-saving capital equipment. Education has reached this stage very late, possibly through a misconception of its task." The classroom, sad to state, is less automated than the kitchen or barn.

The teaching machine can do for educational psychology what the physics, chemistry, and psychological laboratories, for example, have done for their areas. It can serve as an educational laboratory which will eliminate the effects of extraneous variables and permit us to identify the basic instructional variables and their interrelationships; it can provide information about methods of teaching free of uncontrolled variations due to differences among teachers. Automated devices will bring us closer and closer to a true science of learning, and, accordingly, to a true technology of education. Education is a technology in the same sense that engineering is a tech-

nology, and psychology and physics, respectively, are the basic sciences of these technologies.

While automated instruction will no doubt turn out somewhere between the expectations of its fiercest enemies and those of its most enthusiastic protagonists, industry and the armed forces are not sitting back waiting for final answers — there never are any. Machines are teaching electronics, the maintenance of complex equipment in the Air Force; and industrial firms are using programmed textbooks and devices in employee-training programs. Schools all over the country are experimenting with self-instructional programs in printed and in hardware form. Programs are being turned out at a rate that has outrun verified programming techniques. Machines ranging in price from about $20 to $5,000 are already on the market. A relatively small number of programs meant for school use are presently commercially available, and many more are in the immediate offing, but not all are interchangeable and few are complete.

None of the traditional modes of instruction, with or without conventional aids, meets the communication and control requirements which inhere in auto-instructional programs and automated devices. However, problems and unanswered questions attend anything that is new. We have a potentially powerful tool and we need to learn how best to use it. Perhaps the future will verify Lloyd Homme's amusingly put prophecy: "In fact, I will go so far as to predict that classrooms of the future, their walls lined with exotic machines, will resemble nothing so much as the emporiums of Las Vegas. I am even willing to bet that the players will be equally intense in their pursuit of reinforcements."

REFERENCES

1. Lumsdaine, A. A. and Glaser, R. (Eds.) *Teaching Machines and Programmed Learning: A Source Book*. Washington, D. C.: National Education Association (Dept. of Audio-Visual Instruction), 1960, xii, 724 pp.
2. Skinner, B. F. The Science of Learning and the Art of Teaching. *Harvard Educational Review*, (Spring) 1954, 24, 86-97. (Reprinted in Ref. 1, pp. 99-113).
3. Skinner, B. F. Teaching Machines. *Science*, 1958, 128, 969-977. (Reprinted in Ref. 1, pp. 137-158).
4. Stolurow, L. M. *Teaching By Machine*. Cooperative Research Project Monograph. Washington, D. C.: Government Printing Office, 1961.
5. West, L. J. Teaching Machines: New Classroom Aids. *Southern Illinois Business Bulletin*, (Summer) 1960, 10, 3-10.

Dr. Lawrence M. Stolurow is Professor in the Department of Psychology and in the Institute for Research on Exceptional Children at the University of Illinois. His two-part article on "The Psychology of Skills" is a notable earlier contribution to the *DPE Journal* (April and June, 1959).

Dr. West is Associate Professor of Business Education in the School of Business at Southern Illinois University and this journal's associate editor for research. He is a frequent contributor of research articles to this and other professional journals.

ERNEST Z. ROTHKOPF

A Do-It-Yourself Kit for Programmed Instruction*

EDITOR'S NOTE: *Dr. Rothkopf has been working with "teaching machines" since 1955. At present, his primary research efforts are concerned with human verbal learning. He is convinced that a better understanding of the fundamental laws of human verbal learning is necessary in order to provide a satisfactory rational basis for programmed instruction. In the excerpt reprinted here Dr. Rothkopf suggests that, in the light of our present knowledge about programming techniques, teachers can be effective programmers, and he explains how they might go about writing self-instructional programs.*

. . . It is difficult to train programmers because programming at present lacks in science, and it is difficult to build up a cadre of professional, general program writers because considerable subject matter knowledge and sophistication is required. These considerations are critical for the continued development of programmed instruction as an educational tool. Programs are needed in considerable numbers if this new instructional technique is to get the kind of widespread try-outs which appear warranted by its promise. Further, the easy availability of programs in a variety of different subject matters will stimulate the kind of study and research which is likely to lead to improvements in the rational basis for programmed instruction and instruction in general.

THE TEACHER AS PROGRAMMER

These difficulties suggest to me that we should look to the teacher as a supplier of programmed instructional materials. Experienced teachers know a good deal about explaining. They know the language level at which their students operate. They are for the most part bright, articulate, and patient. Teachers are familiar with those parts of their courses which usually offer difficulty — those which require careful staging and slow progress. Most teachers have learned what the main sources of difficulty in their courses are and a number of verbal maneuvers or working examples which nearly always bring the sunrise of understanding into their pupils' faces. What I propose is that teachers put information of this kind to work in writing programs.

*Excerpted from the *Teachers College Record*, Volume 62, No. 3, (December 1960), pages 195-201, and reprinted with the permission of the author and the editor.

WRITING A PROGRAM[1]

The first step in attempting to write a program is to look at some already available. Commercial programs are just now coming on the market. Nearly all of these can be used in booklet form and do not require machines. Some good available examples of extrinsic programs in the general style of Skinner have been produced through the efforts of a small group of experimental psychologists in New Mexico *(8, 9, 10, 11)*. Good examples of intrinsic programs are also on hand *(2, 3, 4, 6)*. Read several of these or similar programs carefully, and try to get a feeling for the way in which these programmers organize and develop their subject matter.

Once you have an appreciation of the technique, try to prepare a program yourself. Pick some reasonably small, relatively independent, and self-contained unit within a course. Wait until you have had much more experience and have gained confidence in your ability before you undertake more ambitious jobs such as programming the major portions of a course. Otherwise you may become discouraged. Suitable topics include Ohm's law, the reigns of British monarchs from Elizabeth to the present, the varieties of verse meter, square roots and exponents, or the concept of chemical valence. It is a good idea if several teachers dealing with the same subject in the same school try programming the same unit. This procedure creates a spirit of competition which may be helpful in writing programs. Having several teachers attempt to program the same subject matter also creates a group of people who would be well motivated to criticize each other's work.

In attempting to write programmed materials it is well to dispel the notion that you are engaging in some arcane enterprise. Imagine what you would have to tell or show a single student in a tutorial situation in order for him to meet all course requirements without having to read a textbook. What would be a good way to organize the information? Is the material primarily conceptual or factual? If the material is chiefly conceptual it is probable that there are several different conceptual structures that will support the kind of performance you want in your student. A pupil can learn enough electricity to solve simple DC circuit problems in a number of ways. You can teach him about sub-atomic particles and forces. All subsequent matter can be developed in these terms. On the other hand, atoms need never be mentioned. The notion of electron flow can be used to develop the main relevant ideas. Even more simply, the student can be taught a series of rules about various electrical phenomena. Students taught according to these various conceptual systems may perform differently at the end of the course with respect to certain kinds of requirements. The "rule" and the "electron flow" students may have subsequent difficulties in learning to understand new relationships between electron flow and the electrolytic dissociation of solutions. But the performance of all students is likely to be nearly identical as far as DC circuit problems are concerned. Choose the conceptual system which you want to support your students' performance on the basis of simplicity consistent with need, transfer value for other portions of the course or other courses, and mnemonic value.

Once a particular conceptual framework has been chosen, make an ex-

[1]No effort is made to document the basis for the recommendations which are enumerated in the following paragraphs. Some of these probably cannot be defended in terms of any single generally accepted empirical fact. They are merely my opinion and those of a few other workers in this area with whom I have had contact. The recommendations seem safe and plausible aids to programming instructional material.

haustive list of the major conceptual terms within this framework. All terms which stand in nontrivial relationships to other terms should be listed in this manner. Consider the most basic relationships among the listed terms, and organize the development of your material accordingly. Remember that the conventional way of organizing your topic may not be the best. Most standard textbooks use the style of exposition and the organization of the scholarly works on which they are based. Seldom are they written specifically to promote learning!

If your subject matter is primarily factual, consider whether you can provide some sort of mnemonic framework which will help to organize the facts. "Eras" in history are examples of such mnemonic organizers. The mnemonic framework should tend to break down the required mass of facts into no more than five small clusters or units of facts (in the example of the monarchs of Britain, this could be the several dynasties) which can readily be handled by the student. The clusters may have subordinate units (preferably no more than five, like kings within dynasties). If the material warrants it, several levels of subordinate units should be constructed.

EXTRINSIC VS. INTRINSIC

Now decide whether you want to use the linear progression of extrinsic programming or the branching format of the intrinsic method. It is safer to stick to extrinsic programming unless students' error tendencies can be easily anticipated and errors fall into a small number of nicely differentiated categories. Also, intrinsic programming usually requires multiple-choice responses, and these require more effort to construct than the fill-in items which are commonly used in the extrinsic method. Regardless of the method you choose, prepare your program in book form. Although some inexpensive teaching machines are now on the market, it is more economical to start with small mimeographed booklets. Put one frame on a page and the answers to questions or incomplete sentences on the reverse side of the question leaf. Do not worry about your student's "cheating" by looking ahead in the booklet. It is not clear whether this kind of "cheating" interferes with learning at all.

Use your list of conceptual relationships and your conceptual or mnemonic outline in preparing the individual instructive frames. Each relationship on your list should be covered by at least two frames and probably more. Cover all syntactic and semantic variations you can conceive for each relationship. For example, in preparing items covering a given formula in physics, use all permutations (algebraic variations) of the formula. Include not only the permutations of the various symbols, but also the words which stand for each of the symbols. Substitute each word or phrase, singly or in combinations into the formula. Try restating the formula in sentence form in several different ways. These restatements of relationship do not have to appear in immediate sequence in the program but should be reasonably close together. It is thought that such restatements in various form are a very efficient form of repetitive practice. Skinner (7) has characterized this procedure picturesquely as "ringing the changes."

In preparing each individual frame, satisfy yourself that the student is likely to be familiar with all the words you are planning to use. If there are some doubtful terms, write a special frame in which you define each one. Each frame in the extrinsic method should consist of one or two simple

sentences which carry the exposition forward in *very small steps*. One method is to compose a frame of a modest amount of old information and a *very small* amount of new material. A good strategy is to state some relationship or rule and then illustrate by an example.[2] Then restate the rule or provide another example, but omit one or more terms in this frame. The student is asked to provide the missing words. Some programs have a missing term in nearly every frame. It is thought to be good procedure to write incomplete statements or questions in such a way as to assure that the student responds to these correctly in the great majority of instances. This can be done by making the question frame wholly or partially synonomous with the most recent instructive items. Some program writers have even used prompting strategems of various kinds to increase correct responses. These have included devices such as providing some of the letters of the missing terms or various mnemonic rhyme schemes.

STUDENT TRY-OUTS

Write your program on 3-by-5 cards. This will facilitate rearranging the order of the frames and inserting new frames in the sequence when they are needed. As soon as you have completed a first draft, rewrite it right away. Novice programmers often tend to put too much new information into their frames. Aim at having twice as many frames in your second draft as you had in your first draft of the program. If the second draft seems adequate, try your deck of 3-by-5 cards on one of your students. Watch your student while he works. Record the amount of time he spends on each frame. Encourage him to ask you questions if, *and only if*, he runs into difficulties. He will quickly find any undetected weak portions of the program for you. Continue this general procedure until you feel reasonably sure that the program you have written will teach your students what you want them to learn. Then find out whether you have done a good job by administering the program to the students in your classes and testing them. *Remember that you have not written an instructional program unless the program teaches all your students and teaches them well.* If your effort does not meet this standard, then try to improve it or write it off as a useful exercise.

It does not seem unreasonable that teachers who are responsible for the same courses within some locality — e.g., a regional school system — could evolve some informal organization which would select from their own programming efforts a program which would suit their needs. The program so selected could then be produced in booklet form by a commercial printing firm. If the school system is adventurous, some effort can be made to shop among various commercial sources for a "teaching machine" appropriate to the requirements of the programs. Eventually, the entire production process for programs, including writing, will probably be handled by commercial organizations, perhaps book publishers. The experience which individual teachers gain in writing programs is likely to be of help to them later in evaluating available commercial programs.

There is another advantage which the teacher could derive from the experience he gains in programming. Commercial producers of programs may someday require good programmers in fairly large numbers and may have to pay them handsomely. The teacher who finds that he is good at program-

[2]This is one of the combinations used in the so-called "Ruleg" method; for further description of this method see Homme & Glaser (5).

ming may want to consider the possibilities of professional programming as an interesting and profitable career.

SUMMARY

I have attempted to argue that the development of programmed instruction as an educational tool is at present impeded by two difficulties. These difficulties grow from the weak rational basis for program writing and from inadequate subject matter knowledge among program writers. One interim solution is to look to the school teacher as a source of programs. If the teacher meets this challenge and becomes interested and skillful in programming, we may expect a massive increase in the number and variety of available programs. Regardless of the source, a large variety of programs is needed if programmed instruction is to get the kind of extensive trials which its promise as an educational tool seems to warrant. Further, the availability of a great variety of programs is likely to stimulate the study and research which may contribute to the rational basis for programmed instruction and the instructional process in general.

REFERENCES

1. Crowder, N. A. Automatic tutoring by means of intrinsic programming. In E. Galanter (Ed.) *Automatic teaching: The state of the art.* New York: Wiley, 1959, pp. 109-116.
2. Crowder, N. A. *The arithmetic of computers.* New York: Doubleday, 1960 (in press).
3. Crowder, N. A., & Martin, G. C. *Adventures in algebra.* New York: Doubleday, 1960.
4. Goren, C. H. *The elements of bridge.* New York: Doubleday, 1960.
5. Homme, L., & Glaser, R. Relationships between the programmed textbook and teaching machines. In E. Galanter (Ed.) *Automatic teaching: The state of the art.* New York: Wiley, 1959, pp. 103-108.
6. Hughes, R. J., & Pipe, P. *Introduction to electronics.* New York: Doubleday, 1960.
7. Skinner, B. F. Teaching machines. *Science,* 1958, 128, 969-977.
8. Teaching-Machines, Inc. *Descriptive statistics.* Albuquerque, N. M.: Teaching Machines, Inc., 1960.
9. Teaching Machines, Inc. *Fundamentals of algebra.* Albuquerque, N. M.: Teaching Machines, Inc., 1960.
10. Teaching Machines, Inc. *Fundamentals of music.* Albuquerque, N.M.: Teaching Machines, Inc., 1960.
11. Teaching Machines, Inc. *Fundamentals of spelling.* Albuquerque, N. M.: Teaching Machines, Inc., 1960.

Dr. Rothkopf is a Research Psychologist with the Bell Telephone Laboratories at Murray Hill, New Jersey. He is a frequent contributor to psychological journals.

DELTA PI EPSILON fraternity, founded in 1936, is an honorary graduate organization for men and women devoted to the advancement and professionalization of business education. Through its ideals of service, leadership and cooperation, the fraternity strives to make significant and unique contributions to professional growth and scholarly achievement in business education. In the words of its founder, Dr. Paul Lomax, can be seen the scope of the fraternity, "The professional interests of Delta Pi Epsilon encompass the whole of business education in relation to the entire fields of American business and American education. Its membership . . . must always think in terms of the common good and advancement of all our business teachers and of all students who pursue courses in business education."

CURRENT CHAPTERS:

ALPHA: New York University
BETA: Oklahoma State University
GAMMA: University of Pittsburgh
DELTA: University of Cincinnati
EPSILON: Boston University
ZETA: Women's College, University of North Carolina
ETA: University of Denver
THETA: Indiana University
IOTA: Syracuse University
KAPPA: University of Michigan
LAMBDA: Northwestern University
MU: University of Tennessee
NU: University of Kentucky
XI: University of Florida
OMICRON: State University of Iowa
PI: Ball State Teachers College
RHO: Ohio State University
SIGMA: University of Oklahoma
TAU: Columbia University
UPSILON: University of Mississippi
PHI: University of Minnesota
CHI: Pennsylvania State University
PSI: University of Southern California
OMEGA: George Peabody College for Teachers
ALPHA ALPHA: Colorado State College
ALPHA BETA: University of Illinois
ALPHA GAMMA: University of Houston
ALPHA DELTA: Kansas State Teachers College (Emporia)
ALPHA EPSILON: North Texas State College
ALPHA ZETA: Temple University
ALPHA ETA: University of Wisconsin
ALPHA THETA: University of Texas
ALPHA IOTA: University of Colorado
ALPHA KAPPA: San Francisco State College
ALPHA IOTA: University of Colorado

NATIONAL OFFICERS:

FRANK HERNDON, *President*
ROBERT LOWRY, *Vice-President*
MARY BATES, *Secretary*
JOHN BINNION, *Treasurer*
IRENE PLACE, *Historian*

DELTA PI EPSILON
National Honorary Graduate Fraternity in Business Education
Ruth I. Anderson, Executive Secretary
North Texas State College
Denton, Texas

RETURN POSTAGE GUARANTEED